Department Of Mathematics
University Of Texas At Austin
Austin, Texas, United States Of America

`lazar_ilic@yahoo.com`

Expected Value Of Minimal Transversal Across Permutations

Lazar Ilic

Abstract

Given a permutation of a set of $n^d = m$ elements embedded in a hypercube/tensor lattice of length n and dimensions d, we consider computing the minimum of the maximum element in a minimal hitting set, a set of n vertices such that there exists precisely 1 vertex in each of the nd hyperrows/fibers. Furthermore, we consider the minimum of the sum of elements in a minimal hitting set. We refer to these as MiniMax and Minimum Sum. These queries arise naturally in the context of a variety of domains, including the tasks of the pre eminent William Lowell Putnam Mathematics Competition 1959 B4 and the United States Of America 2020 2. Slightly puzzling puzzles arise in certain special subcases and further literature references abound.

Mathematics Subject Classifications: 05C88, 05C89

1 Introduction

The Putnam task asks: Given the following matrix:

$$\begin{bmatrix} 11 & 17 & 25 & 19 & 16 \\ 24 & 10 & 13 & 15 & 3 \\ 12 & 5 & 14 & 2 & 18 \\ 23 & 4 & 1 & 8 & 22 \\ 6 & 20 & 7 & 21 & 9 \end{bmatrix}$$

Choose 5 of these elements, no 2 from the same row or column, in such a way that the minimum of these elements is maximal. We might equivalently have inverted the matrix in the sense of taking 26 minus each element that is to say considering the matrix $26I - A$ and choosing 5 such elements that the maximum of the selected elements is minimal.

$$\begin{bmatrix} 15 & 9 & 1 & 7 & 10 \\ 2 & 16 & 13 & 11 & 23 \\ 24 & 21 & 12 & 24 & 8 \\ 3 & 22 & 25 & 18 & 4 \\ 20 & 6 & 19 & 5 & 17 \end{bmatrix}$$

In any case, a canonical solution to pencil and paper this task might be to simply eyeball that upon inspection the set $[1, 2, \ldots, 11]$ works with the choice $(1, 11, 8, 3, 6)$, however the set $[1, 2, \ldots, 10]$ fails to induce and produce such a set. This brings us immediately into the subcase we will be paying most attention to in this paper, where the underlying set of which is being permuted is $[n^d] = [1, 2, \ldots, n^d]$. And here in this case the puzzle of the dimension $d = 2$ subcase. We could opt to permute rows and columns such that without loss of generality, the 1 is located in the first index, but opt not to, rather considering quantities over all possible permutations.

It ought to be noted that in reality and possibly, context dependent, in some of these computations, there exist approximation algorithms for quantities and distributions based upon relatively naive simple random sampling which produce fairly performant implementations rather than directly enumerating combinatorially, or even with half performant combinatorial executions in computer algebra software.

2 The $d = 2$ Subcase For MiniMax

This section describes background information about algorithms.

Theorem 1. *There exists an $O(n^{\frac{5}{2}} \cdot \log(n))$ algorithm for producing the MiniMax, or minimum maximum element over all minimal hitting sets for a given permutation.*

Proof. Indeed, considering this as a classical special instance of Maximum Flow where there is a bipartite graph with a set of vertices corresponding with the rows and a set of vertices corresponding with the columns. The relevant edges between a row vertex and a column vertex being associated with the value at that unique square in the $n \times n$ grid. With an auxiliary source node leading into, having a single edge of capacity 1 to each of the rows vertices, and a sink node having edges of capacity 1 from each of the column vertices. It then is enough to binary search for a threshold value, as in the earlier pencil and paper example. Binary search for a minimal threshold value wherein the Maximum Flow induced is indeed n. However, the Harpcroft-Karp-Karzanov algorithm for Maximum Bipartite Matching is a classic which outperforms for such a special instance and may be utilised here for example. $\square$

Theorem 2. $n \leqslant$ *MiniMax* $\leqslant n^2 - n + 1$

Proof. A lower bound and upper bound arise immediately and naturally from degenerate input cases constructions. In particular, if the main diagonal consists of the elements $1, 2, \ldots, n$ then we may select that set. This bound is obtained in general if and only if that subset of/in the permutation induces a hitting set. And if the permutation is the most simple listing of all of the elements in order say across the rows first and then down the columns, then the $n^2 - n + 1$ bound is obtained. But in general if and only if the elements $n^2 - n + 1, n^2 - n + 2, \ldots, n^2$ are all in the same row or column, thus forcing the selection of 1 of them by the Pigeonhole Principle. It is worth considering the distribution of the maxima over all induced minimal hitting sets. $\square$

Theorem 3. *The Expected Value of the MiniMax is $\omega(n)$.*

Proof. One of the first natural queries to ask is if there exists a very simple proof say that the Expected Value for the MiniMax will be $O(n), O(n\log(n)), O(n\sqrt{n})$. And one could even run simulations say over pseudorandom permutations, probabilistically convering on the true expected value mean over sufficiently many iterations by the discrete domain of support and the Central Limit Theorem. But in particular in this case, as in the Putnam task motivating example, one can execute this with the relevant binary grids looking to gain information on just how many elements are usually needed to induce a hitting set. Of course an upper bound may arise from an invocation of the Probabilistic Method. Namely, in the growing support for the number of invoked minimal hitting sets one can consider a uniformly random distribution of say a of the 1s elements and sum over each hitting set, in this case from the earlier map correspondence, permutations on $[n]$. Then as the probability for each appearing is $\prod_{j=1}^{n} \frac{a+1-j}{n^2+1-j}$ one notes that $\lim_{n\to\infty} n! \prod_{j=1}^{n} \frac{f(n)+1-j}{n^2+1-j} = \infty$ for some suitable $f(n)$ which is $O(n)$ due to Stirling's Approximation here providing the needed $n^n\sqrt{n}$ asymptotic to clear in conjunction with an n^n asymptotic in the numerator over the n^{2n} asymptotic emerging from the denominator. Now this does not actually ensure the desired, as we need further commentary on the degree to which the induced cases are disjoint, e.g. there are not simply many 0 cases and some others which produce a high number. However, the implicit structure here lends intuitive credence when one consider the distribution and the fact that additional inductions seem strictly less likely in a sense, perhaps a hypothesis on unimodality or weak monotonicity is assured.

Actually, this hand wave is wrong and False. See later for now. $\qquad\square$

Theorem 4. *We attempt to compute the formal literal formulae for the distribution using standard combinatorics techniques. And in the process generate potential novel sequences for the Online Encyclopaedia Of Integer Sequences. Via direct programmatic counting.*

Proof. It would suffice to more simply count the number of binary tensors with a certain number j of 1s entries that do contain an induced minimal hitting set. Because we could then multiply this count by the appropriate $j!(m-j)!$ term to count the number of permutations such that the MiniMax was $\leqslant j$. And then we could simply subtract off, e.g. take the consecutive terms Deltas to produce the literal count of permutations which had that precise MiniMax value e.g. by having the subset $[1, 2, \ldots, j-1]$ not inducing a minimal hitting set but having the subset $[1, 2, \ldots, j]$ do it. And thus it suffices to invert and count the number of such binary tensors which do not induce a single minimal hitting set e.g. induce 0 by avoiding all of the potential hitting sets.

Principle Of Inclusion And Exclusion will not work easily here in general.

It ought to be noted that the number of subsets which are minimal hitting sets can be counted rather easily. Indeed, we may select the elements of such a subset in order and divide out by the relevant factorial so obtain $\frac{1}{n!} \cdot \prod_{j=1}^{n}(n+1-j)^d = \frac{1}{n!}(n!)^d = (n!)^{d-1}$.

Furthermore, one representation for the number of minimal hitting sets contained in a particular binary tensor comes from a Stack Math Overflow thread "Multi-dimensional permanent" by the user and friend Alex Meiburg by the user Anna Taranenko:

Indeed, there is no generally accepted definition of the permanent of multidimensional tensor (or matrix) yet. I suppose that the basic definition of the permanent of a d-dimensional matrix A of order n is the sum over all permutations $\sigma_1, \ldots, \sigma_{d-1} \in S_n$ of products $\prod_{i=1}^{n} a_{i,\sigma_1(i),\ldots,\sigma_{d-1}(i)}$. I also mention that the paper https://arxiv.org/pdf/1101.3629.pdf uses the same definition of the permanent.

Concerning hypergraphs, the permanent of a d-dimensional $(0,1)$-matrix could be interpreted as the number of perfect matchings in a d-uniform d-partite hypergraph and it is connected with the number of perfect matchings in a general d-uniform hypergraph. It is also related to the number of some combinatorial structures, for example, to the number of transversals in latin squares and hypercubes.

I know no relation between multidimensional permanent and determinant (possibly, because I know no satisfactory definition of the multidimensional determinant too).

s-permanent from http://www.sciencedirect.com/science/article/pii/0024379587903119 for $s = 1$ coincides with the previous definition. For $s = d - 1$ the s-permanent of a d-dimensional matrix A of order n is the sum over all sets D of n^{d-1} entries such that each line (1-dimensional plane) of A contains exactly one element from D of products $\prod_{a_{i_1,\ldots,i_d} \in D} a_{i_1,\ldots,i_d}$. For $1 < s < d - 1$ the s-permanent of a d-dimensional matrix may not exist. It will be interesting what does the s-permanent mean for hypergraphs.

Some additional information about multidimensional permanents you can find in https://link.springer.com/article/10.1134/S1990478916040141

So we are looking for the number, for each possible number of 1s entries, of binary tensors with a nonzero such permanent. Of course a general direct brute force approach with say Ryser's Algorithm will work here to produce the target output but we expect that superior combinatorial techniques will at the least reveal tight asymptotics for the numerics involved.

In any case we produce the sequences of numbers of such binary tensors which do induce more than 0, that is to say $\geqslant 1$, at least 1 minimal hitting set:

$0, 1$

$0, 0, 2, 4, 1$

$0, 0, 0, 6, 36, 81, 78, 36, 9, 1$

$0, 0, 0, 0, 24, 288, 1512, 4464, 8010, 9312, 7432, 4272, 1812, 560, 120, 16, 1$

$0, 0, 0, 0, 0, 120, 2400, 22200, 124800, 471450, 1257840, 2463200, 3665500, 4276600, 4017800, 3102040, 1992$

Further this becomes the sequences of the number of permutations such that the indices at the values of the prefix $[a] = [1, 2, \ldots, a]$ induce such a minimal hitting set:

$0, 1$

$0, 0, 8, 24, 24$

$0, 0, 0, 25920, 103680, 233280, 336960, 362880, 362880, 362880$

$0, 0, 0, 0, 275904921600, 1379524608000, 3950456832000, 8164277452800, 13021876224000, 1703085834240$

$0, 0, 0, 0, 0, 3503378891774361600 0000, 2102027335064616960 00000, 716348789186494464 000000, 178979$

And then taking the Deltas or consecutive differences this becomes the precise number of permutations which have this MiniMax value.

0, 1
0, 0, 8, 16, 0
0, 0, 0, 25920, 77760, 129600, 103680, 25920, 0, 0
0, 0, 0, 0, 275904921600, 1103619686400, 2570932224000, 4213820620800, 4857598771200, 4008982118400,
0, 0, 0, 0, 0, 35033788917743616000000, 175168944588718080000000, 506146055680032768000000, 107344?

And thus produce the Expected Values of:

1

2.6666666666666667

5.071428571428571

8.003796203796204

11.317214605140302

And sums of:

1

64

1840320

167461746278400

17554369284578436710400000   □

3   The $d = 2$ Subcase For Minimum Sum

Theorem 5. *There exists an $O(n^3)$ algorithm for producing the Minimum Sum, or minimum sum of elements over all minimal hitting sets for a given permutation.*

Proof. Again as before, we identify with a bipartite graph and an instance this time of Minimum Cost Maximum Bipartite Matching wherein each edge is present and has a cost of the value in that row column position between that row vertex and that column vertex. And so the Hungarian algorithm applies.   □

Theorem 6. $\frac{n^2+n}{2} \leqslant$ *Minimum Sum* $\leqslant \frac{n^3+n}{2}$

Proof. A lower bound and upper bound arise immediately and naturally from degenerate input cases constructions. In particular, if the main diagonal consists of the elements $1 + 2 + \cdots + n = \frac{n(n+1)}{2} = \frac{n^2+n}{2}$ then we may select that. And if the permutation is the most simple listing of all of the elements in order say across the rows first and then down the columns, it is a classic puzzle sort of in fact that the sum will be the same across all minimal hitting sets, namely $\frac{n(n^2+1)}{2} = \frac{n^3+n}{2}$. So $\frac{n^2+n}{2} \leqslant$ Minimum Sum $\leqslant \frac{n^3+n}{2}$. It also ought to be noted here that in the $d = 2$ case, a minimal hitting set itself corresponds with a permutation of $[n] = [1, 2, \ldots, n]$, for example as the index of the selected element in the ith row. And that this upper bound construction can be seen as the uniform point mass distribution of the sums over all induced minimal hitting sets.   □

Theorem 7. *The Expected Value of the Minimum Sum eh $O(n^2)$.*

Proof. One of the first natural queries to ask is if there exists a very simple proof say that the Expected Value for the Minimum Sum will be $O(n^2), O(n^2 \log(n)), O(n^2\sqrt{n})$. And

one could even run simulations say over pseudorandom permutations, probabilistically convering on the true expected value mean over sufficiently many iterations by the discrete domain of support and the Central Limit Theorem. Here however we may use the previous part to immediately give us the $O(n^2)$ asymptotic as indeed in each subcase instance of MiniMax we had the same probabilities associated with the underlying permutations and can simply upper bound the sum over that minimal hitting set with n copies of the maximal element in it, or the MiniMax, and thus upper bound this quantity with n times that quantity e.g. Minimum Sum $\leqslant n$ MiniMax.

Eh maybe wrong for now see later. $\qquad\square$

Theorem 8. *We attempt to compute the formal literal formulae for the distribution using standard combinatorics techniques. And in the process generate potential novel sequences for the Online Encyclopaedia Of Integer Sequences. Via direct programmatic counting.*

Proof. $\qquad\square$

4 The $n = 2$ Subcase For MiniMax

Note here that in the $n = 2$ subcase, the minimal hitting sets are simply pairs e.g. in a pseudorandom matching induced on a permutation of $[2^d]$. And that some of these permutation statistics will work out similarly in a general context setting of an even case.

Theorem 9. *There exists a simple $O(2^d)$ algorithm for producing the MiniMax value.*

Proof. A simple algorithm is to simply process through the entire permutation as input and produce the target output. However, if we had the ability to both query the value at an index and the index of an element, say these 2 are both read or taken in a function as input, then we could very simply execute a sort of a 2 pointers approach or rather process upwards querying the partner of $1, 2, \ldots$ in sequence and storing the minimum partner seen thus far until terminating when we halt upon reaching that now value from this lower left hand side. In such a theoretical setting where both of these query operations are taken to be $\Theta(1)$ operations then this algorithm terminates after MiniMax steps and thus has expected runtime as below. $\qquad\square$

Theorem 10. $2 \leqslant$ *MiniMax* $\leqslant 2^{d-1} + 1$

Proof. Indeed it is trivial to see that these occur if and only if the 1 is paired with the 2 and for the upper bound if and only if none of the elements $[2^{d-1} + 1, 2^{d-1} + 2, \ldots, 2^d]$ are paired together e.g. the matching is from the subsets $[1, 2, \ldots, 2^{d-1}]$ with $[2^{d-1} + 1, 2^{d-1} + 2, \ldots, 2^d]$. $\qquad\square$

Theorem 11. *Sum Of MiniMaxes* $= 4^{2^{d-1}}((2^{d-1})!)^2$

Proof. See below for the formal derivation of the distribution. One then obtains the summation via for example the usual routes mediated by computer algebra software:

$$\sum_{j=2}^{2^{d-1}+1} (j-1) \cdot \binom{\frac{2^d}{2}}{j-1} \cdot (j-1)! \cdot 2^{j-2} \cdot (2^d - j)! = \text{ Sum Of MiniMaxes } = 4^{2^{d-1}}((2^{d-1})!)^2$$

Expected Value Of MiniMax $= \frac{4^{2^{d-1}}((2^{d-1})!)^2}{(2^d)!}$

This has to do with the Central Factorial Numbers OEIS A002454 as well as related papers like Burke-Schumann Diffusion Flame Theory by F.W. Williams in the subsection Bessel Functions And Their Polynomial Approximation. $\square$

Theorem 12. $P[MiniMax = j] = \frac{(j-1)\cdot\binom{\frac{2^d}{2}}{j-1}\cdot(j-1)!\cdot 2^{j-2}\cdot(2^d-j)!}{(2^d)!}$ *for* $j \in [2, 2^{d-1}+1]$.

Proof. We enumerate combinatorially. It ought to be noted that this particular structure has to do with the Peter Winkler puzzle from Mathematical Puzzles titled Random Intervals. There it was shown that the probability the intervals induced by such a random permutation contains an interval which intersects all of the others is $\frac{2}{3}$. But in any case, I digress. Here it may be seen that for the MiniMax to be j then it must be the case that j paired off with some value of $< j$ and all of the other values $< j$ paired off with a value $> j$. Thus one immediately obtains that $P[\text{MiniMax} = j] = \frac{(j-1)\cdot\binom{\frac{2^d}{2}}{j-1}\cdot(j-1)!\cdot 2^{j-2}\cdot(2^d-j)!}{(2^d)!}$ for $j \in [2, 2^{d-1}+1]$. Indeed this is the number of ways of selecting locations for the first $j-1$ values which must be in separate pairs followed by a simple permutation to finish it all off. $\square$

5 The $n = 2$ Subcase For Minimum Sum

In this case as well one may consider pairs as before.

Theorem 13. *There exists a simple* $O(2^d)$ *algorithm for producing the Minimum Sum value.*

Proof. Again as before we may simple process through the entire input storing the minimum thus far until the end for output. However, if equipped with those $\Theta(1)$ lookup operations we may of course again process through in a quasi 2 pointers simple style that is to say process from left or least values increasing, storing the minimum sum thus far until we hit $\frac{1}{2}$ of that value in order starting through $1, 2, \ldots$. The expected runtime for this algorithm again is given below implicitly in the Expected Value section. $\square$

Theorem 14. $3 \leqslant$ *Minimum Sum* $\leqslant 2^d + 1$

Proof. This is fairly trivial. Indeed it arises if and only if in this case the pair $1, 2$ appears and for the upper bound we would need to have 1 paired with 2^d, 2 paired with $2^d - 1$, etc. all the way up the chain. Again, this is because the sum over all the pairs, and hence the average value of a sum in a minimal hitting set is indeed this quantity and for it to be the minimum it must be obtained everywhere by every pair. $\square$

Theorem 15. *The expected value and sum may be computed explicitly in terms of the distribution below.*

Proof. See below for the derivation of the formal formulae discrete distribution. We hope to resolve if this summation does or does not permit a nicer representation, and computationally superior implementation. □

Theorem 16. $P[Minimum\ Sum = j] = \prod_{i=1}^{\lfloor \frac{j-1}{2} \rfloor} \frac{2^d - j + 2}{2^d + 1 - 2i} - \prod_{i=1}^{\lfloor \frac{j}{2} \rfloor} \frac{2^d - j + 1}{2^d + 1 - 2i}$

Proof. We enumerate combinatorially. For the minimum sum to be $\geqslant j$ requires that 1 be paired with some element $\geqslant j - 1$, 2 be paired with some element $\geqslant j - 2$, etc. up that chain. And thus one immediately directly obtains that $P[\text{Minimum Sum} = j] = \prod_{i=1}^{\lfloor \frac{j-1}{2} \rfloor} \frac{2^d - j + 2}{2^d + 1 - 2i} - \prod_{i=1}^{\lfloor \frac{j}{2} \rfloor} \frac{2^d - j + 1}{2^d + 1 - 2i}$. Indeed this follows by a standard maths contest style probabilistic argument upon computing via uniform random properties the probabilities of these pairings occuring in sequence. □

Theorem 17. *The Minimum Sum Expected Value asymptotic is easily computable.*

Proof. One can almost certainly produce an easier upper bound via a simple approximation and set a threshold in the limit of the product such that the probability of clearing that threshold goes to 0. A tighter approximation can be computed by taking the logarithm or comparing directly with Stirling's Approximation for the factorial. One aspires to computing this all in a tighter combinatorial way. □

6 The General Case For MiniMax

It ought to be noted that there exist a few natural framings and further readings in these domains. For one, we could consider the dual as an instance of a Minimum Cost/Weight Edge Cover in an underlying hypergraph. Or we can associate a vertex with each hyperrow/fiber and an edge for each entry in the hypercube/tensor connecting that precise corresponding set of hyperrow/fiber vertices. So then the task isomorphs into an instance of Minimum Cost/Weight Vertex Cover in this hypergraph. But again we could also most simply imitate the algorithms of the previous section.

In particular we may simply set up the flow network such that say each entry in the hypercube/tensor has a corresponding vertex on the left side of a bipartite graph and each hyperrow/fiber has a corresponding vertex on the right side. As in the $(n!)^{d-1}$ representation we must first apply a single 1 layer of pre filtering. Then the source must have an edge of capacity d to one of n vertices which prefilters across 1 dimension into each vertex on the left and the edges across the graph must have capacity 1 and cost for example $\frac{\text{Value}}{d}$ to each of the d relevant vertices on the right and then the sink must have edges from each of the hyperrow/fibers with capacity 1. This way an instance of Minimum Cost Maximum Flow will lead to the desired target output value.

In any case there exists a vast body of literature on tensor decompositions, and even some on permutations embedded in tensors. See e.g. Tensor Decompositions And Applications and Provably Efficient Algorithms For Numerical Tensor Algebra. As well as the binary tensors which are quite relevant in the MiniMax case when inducing minimal hitting sets/vertex covers here in this instance.

Theorem 18. *As before.*

Proof. As before. Now the interesting follow on questions of course have to do with approximation algorithms, and parallel implementations relating to memory, lookup, storage. Again, if we read the input in such a way that it is quite easy to query the location/index of the value 1 then this task is very different and can be much more computationally simple especially given how a relatively small, very sub linear portion, or early subset may work out at which point we did not need to further process or query the remaining larger valued elements in the permutation. □

Theorem 19. *As before.*

Proof. As before. □

Theorem 20. *The Expected Value of the MiniMax is $\omega(n)$ it is not $e^{d-1}n$.*

Proof. Need to check these asymptotics computations details re this invocation of Stirling's Approximation for some reason my maths does not seem to be working out right about now.

Certainly one can attempt to imitate argumentations as before. Here however we present a simpler Probabilistic Method argument as in earlier but framed slightly differently. Here we simply compute the probability that a random subset of size n is a minimal hitting set and compute the number of subsets of the selected set, that is the indices locations of the values $[a] = [1, 2, \ldots, a]$ such that in the limit the expected number of contained minimal hitting sets goes to ∞. So we just need that $\binom{a}{n}$ grows faster than the inverse of $\frac{(n!)^{d-1}}{\binom{n^d}{n}}$ is asymptotically $\approx \frac{e^{nd}}{n^{\frac{d}{2}}}$. So it suffices for $a^n = f(n)^n$ to satisfy growing faster than $\approx e^{nd-n} \cdot n^{n-\frac{d-1}{2}}$ which is equivalent with $f(n)$ growing faster than $e^{d-1} \cdot n^{1-\frac{d-1}{2n}}$. But again by simply approximating as in Stirling's Approximation one obtains that an $O(n)$ sized subset works here. In any case again in the limit should go to $e^{d-1}n$ classically. Again as before it should be noted that we are not really taking an approximation for all of these numerator in the denominator terms as n^d in fact it suffices to consider them smoothing to the $n^d - n$ case in conjunction with the classical limit that $\lim_{n\to\infty}\left(1 - \frac{1}{n}\right)^n = \frac{1}{e}$ means $\frac{1}{e^d}$ for a fixed dimension d. And so that is a constant term which factors out when we consider all of the n^{dn} asymptotic terms which cancel and then in particular here we can even get slightly tighter bounds around the estimate in a subconstant term which starts out higher and then asymptotes as $n \to \infty$.

Actually what is ongoing here of course is that in the Principle Of Inclusion for example actually once your subset contains 1 transversal it is actually much more likely and easier for additional elements to induce many many many more and for example with a few alternate sub hitting subsets this can easily blow up factorially so the underlying distribution of number of invoked transverals is actually not quite the way one might initially think and is worth further studying and exploring concrete numerics if possible perhaps.

In particular there is a Probabilistic Method argumentation for this not being enough. Simply based upon the expected number of hyperrow/fibers which are not even hit by

such a random sbuset, which clearly forces the non existence of such a transversal subset. Merely consider that $\lim_{n\to\infty} dn \left(\frac{n-1}{n}\right)^{O(n)} = \infty$. However, this limit is 0 for $\omega(n\log(n))$ so perhaps we can go about re computing some asymptotic here via a half smart in step selection algorithm based upon the expected usual uniform distributions of like missed points and number of incidences in a cancellation step when we choose optimally each time to try and filter down into a transversal. $\qquad\square$

Theorem 21. *Tricky. Can approximate via simulations for small cases as brought up before.*

Proof. Certainly one can attempt to imitate argumentations as before. As mentioned, getting a grasp on this structure with even a Principle Of Inclusion And Exclusion argumentation will be rather tricky. $\qquad\square$

7 The General Case For Minimum Sum

Theorem 22. *As before.*

Proof. As before. $\qquad\square$

Theorem 23. *As before.*

Proof. As before. $\qquad\square$

Theorem 24. *The Expected Value of the Minimum Sum is maybe not $\leqslant \sum_{j=1}^{n} j + \frac{n^d - j + 1 - (n-j+1)^d}{1 + (n-j+1)^d}$. In particular it is maybe not $O(n^2)$.*

Proof. A very weak algorithm would of course be to simply execute a greedy search for an initial upper bound, which could then be further used in downstream searches. Start by selecting the entry with minimal value 1 and then recurse into the induced remainder sub hypercube/tensor. As one is assuming uniform randomness over all of the underlying permutations then one can simply consider the distribution over the next largest element which still remains in the remainder. And so to evaluate the Expected Value of the output of this algorithm which immediately implicit presents an upper bound for algorithms here it suffices to compute the sum of the expected values of each order statistic in our selected set. So e.g. the first value is ensured to be a 1 but the second value could be a 2 or a 3 or even all the way up to a $1 + d(n-1) + 1 = dn - d + 2$ in the worst case scenario. But then I claim this isomorphs into computing the expected value for the minimum element in a random subset of $[j, j+1, \ldots, n^d]$ of size $(n-j+1)^d$. That is to say the remaining sub hypercube/tensor after $j - 1$ steps of our naive greedy selection algorithm. However this is itself yet another classic little puzzley puzzle result of course the expected size of the first gap would be $\frac{n^d - j + 1 - (n-j+1)^d}{1 + (n-j+1)^d}$ leading to an overall expected valuation of $j + \frac{n^d - j + 1 - (n-j+1)^d}{1 + (n-j+1)^d}$ and a sum of $\sum_{j=1}^{n} j + \frac{n^d - j + 1 - (n-j+1)^d}{1 + (n-j+1)^d}$.
    A tighter asymptotic may be obtained with superior combinatorial techniques. Perhaps one could consider the tradeoffs in a simpler selection procedure wherein one say

tests out the next few values and then chooses the one which induces the smallest sum and average over the remainder set after selecting that element. $\qquad\square$

Theorem 25. *OK*

Proof. As before, we aim to generate a closed form, or computationally nice expression for the precise number of permutations which induce that the minimal hitting set has a particular sum. $\qquad\square$

8 Further Comments And Research Directions

Other underlying sets rather than $[m]$ can be explored and in particularly there could exist interesting ones or sets with many real world applications, or important asymptotic performances to ensure in the real world. What if there are measurement errors and only upon selecting the set do we discover the true values? Rather than attempting to compute the minimal hitting set, what if we instead want to partition the entire hypercube/tensor into a disjoint union of n^{d-1} minimal hitting sets. And then evaluate our partition in some way, like perhaps we want for the medians to be as close together bundled up as possible, or perhaps we want the sums to be as close together as possible. Many such natural extensions and avenues of inquiry exist for the freely wandering mind interested in exploring such domains.

9 Acknowledgements

References

[1] B. Bollobás. Almost every graph has reconstruction number three. *J. Graph Theory*, 14(1):1–4, 1990.

[2] J. A. Bondy and R. Hemminger, Graph reconstruction—a survey. *J. Graph Theory*, 1:227–268, 1977. https://doi.org/10.1002/jgt.3190010306.

[3] J. Fisher, R. L. Graham, and F. Harary. A simpler counterexample to the reconstruction conjecture for denumerable graphs. *J. Combinatorial Theory, Ser. B*, 12:203–204, 1972.

[4] E. Hemaspaandra, L. A. Hemaspaandra, S. P. Radziszowski, and R. Tripathi. Complexity results in graph reconstruction. *Discrete Appl. Math.*, 155(2):103–118, 2007.

[5] P. J. Kelly. A congruence theorem for trees. *Pacific J. Math.*, 7:961–968, 1957.

[6] M. Kiyomi, T. Saitoh, and R. Uehara. Reconstruction of interval graphs. In *Computing and combinatorics*, volume 5609 of *Lecture Notes in Comput. Sci.*, pages 106–115. Springer, 2009.

[7] P. K. Stockmeyer. The falsity of the reconstruction conjecture for tournaments. *J. Graph Theory*, 1(1):19–25, 1977.

[8] S. M. Ulam. A collection of mathematical problems. Interscience Tracts in Pure and Applied Mathematics, no. 8. Interscience Publishers, New York-London, 1960.

[9] D. B. West and H. Spinoza. Reconstruction from k-decks for graphs with maximum degree 2. `arXiv:1609.00284vi`, 2016.